IEEE SMC Systems, Man, and Cybernetics Society

THE UNIVERSITY OF BRITISH COLUMBIA Okanagan Campus

IS TIMING CRITICAL TO TRACE RECONSTRUCTION?

Javier Perez Tobia, Apurva Narayan

Why is trace reconstruction important?

- Inaccurate models have severe consequences in safety critical environments.
- Real-life systems have complex dynamics.
- Denoising techniques cannot always be used when analyzing systems.
- Trace reconstruction techniques are one of the best tools to assist in the analysis of these systems.

Related work: Recurrent Neural Networks

Recurrent Neural Networks (RNNs) are the standard approach for handling sequential data.

Proposed improvements:

- Bi-directional LSTM + Grid-LSTM *(Li et.al., 2020)*
- LSTM + Timed-Regular Expression Mining (TREM) or QRE mining *(Sucholutsky et.al., 2019; Mahato et.al., 2020)*

However, most approaches lack a **quantitative notion of time.**

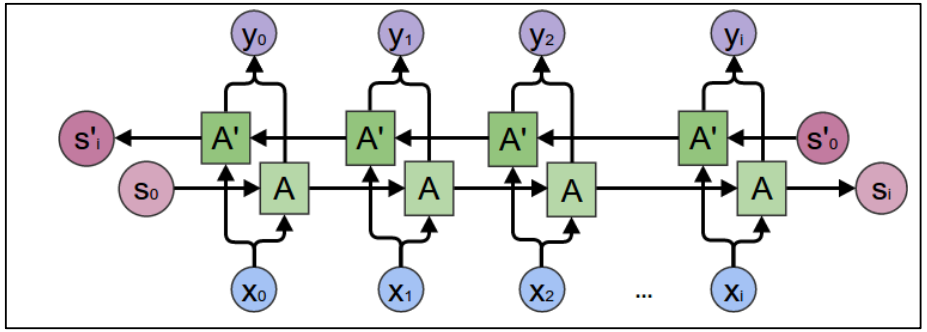

Figure 1: Bi-directional LSTM

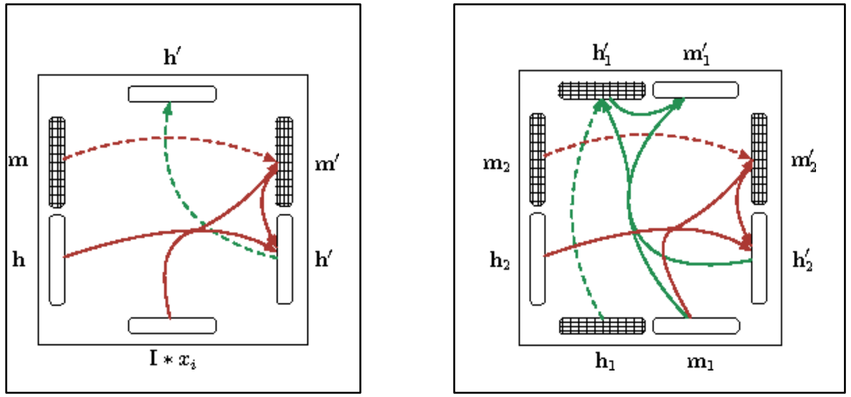

Figure 2: Standard LSTM (left) vs Grid LSTM (right)

Related work: Why is timing important?

Temporal distance between events also matters, not only ordering.

Some approaches that incorporate time:

- Adding time as a weighing factor *(Bailer-Jones, 1998)*
- CT-RNN store memory traces for different time scales *(Mozer et.al., 2017)*
- Time LSTM incorporates time gates in each LSTM unit *(Zhu et.al., 2017)*

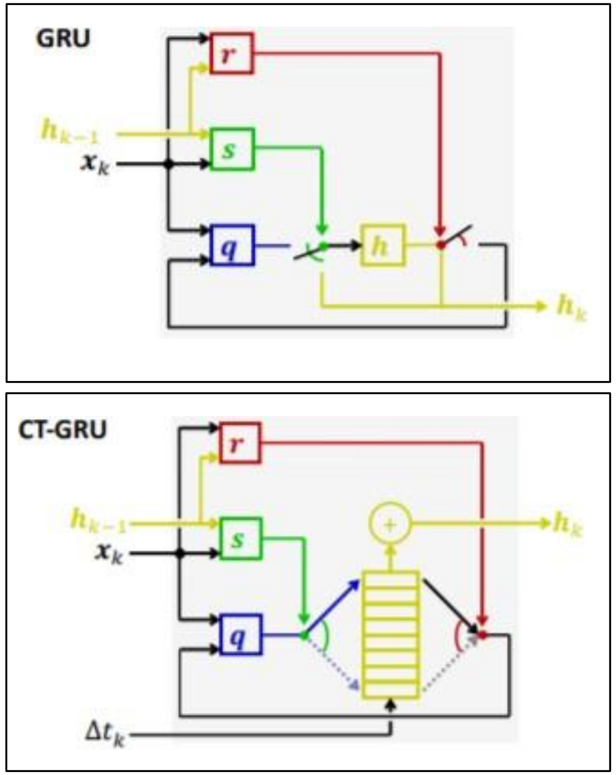

Figure 3: GRU (top) vs CT-GRU (bottom)

Time LSTM *(Zhu et.al., 2017)*

Experiments on recommendation systems reveal improved performance of Time LSTM

Do time gates work as well for other RNN?

Can time gates help in other domains?

		LastFM		CiteULike	
		Recall@10	MRR@10	Recall@10	MRR@10
CoOccur+BPR		0.3217	0.1401	0.6954	0.2901
Session-RNN		0.3405	0.1573	0.7129	0.2997
LSTM		0.2451	0.0892	0.6824	0.2889
LSTM+time		0.2628	0.0977	0.6655	0.2831
Phased LSTM		0.2360	0.0859	0.6087	0.2539
Time-LSTM 1		0.3566	0.1853	0.7428	0.3179
Time-LSTM 2	original	0.3909	0.2250	0.7476	0.3377
	$T1_m = 1$	0.3236	0.1812	0.7058	0.3044
	$T2_m = 1$	0.3643	0.2073	0.7014	0.3105

Table 1: Time LSTM performance on recommendation tasks

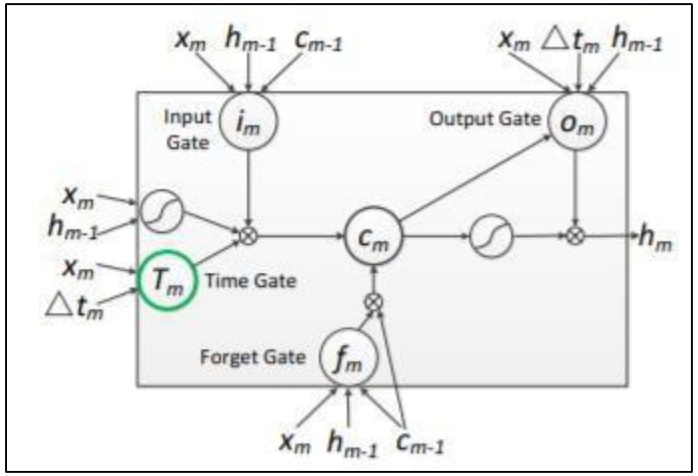

Figure 4: Time-LSTM 1

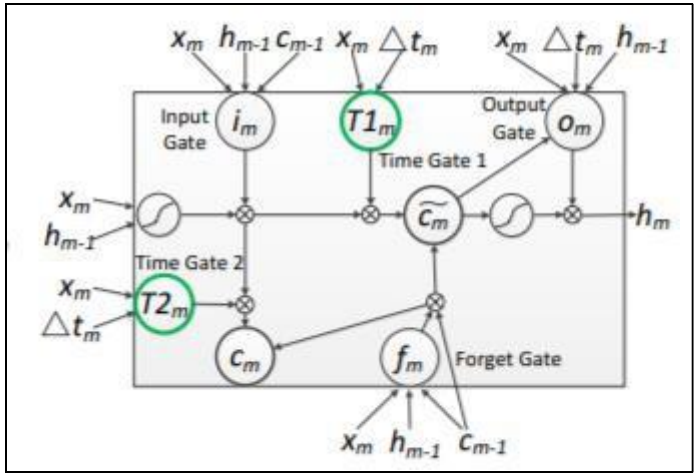

Figure 5: Time-LSTM 2

Time Gated Recurrent Units

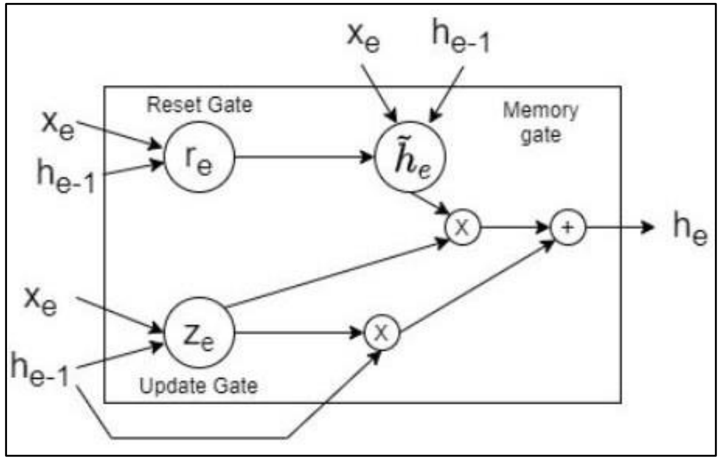

Figure 6: GRU

$$r_e = \sigma_r(x_e W_{xr} + h_{e-1} W_{hr} + b_r) \qquad (1)$$

$$z_e = \sigma_u(x_e W_{xz} + h_{e-1} W_{hz} + b_z) \qquad (2)$$

$$\tilde{h}_e = tanh(x_e W_{x\tilde{h}} + r_e \odot h_{e-1} W_{h\tilde{h}} + b_{\tilde{h}}) \qquad (3)$$

$$h_e = (1 - z_e) \odot \tilde{h}_e + z_e \odot h_{e-1} \qquad (4)$$

Equation 1: GRU

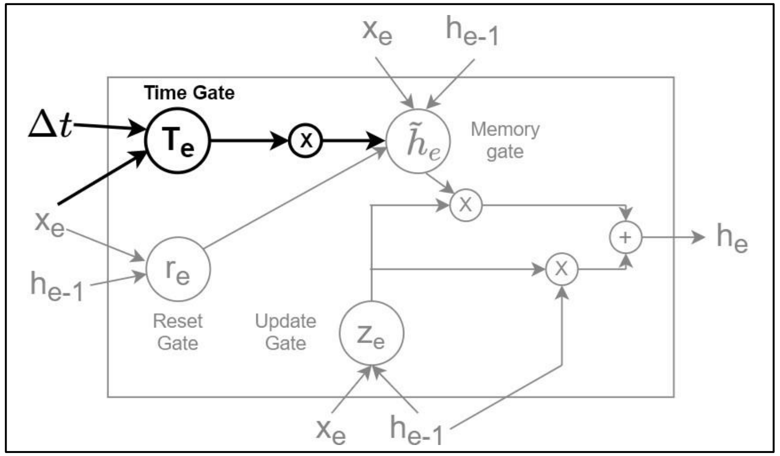

Figure 7: Time GRU

$$r_e = \sigma_r(x_e W_{xr} + h_{e-1} W_{hr} + b_r) \qquad (5)$$

$$z_e = \sigma_u(x_e W_{xz} + h_{e-1} W_{hz} + b_z) \qquad (6)$$

$$\mathbf{T_e} = \sigma_\mathbf{t}(\mathbf{x_e W_{xt}} + \sigma_{\mathbf{\Delta t}}(\mathbf{\Delta t_e W_{tt}}) + \mathbf{b_t}) \qquad (7)$$

$$\tilde{h}_e = \mathbf{T}_e \odot tanh(x_e W_{x\tilde{h}} + r_e \odot h_{e-1} W_{h\tilde{h}} + b_{\tilde{h}})(8)$$

$$h_e = (1 - z_e) \odot \tilde{h}_e + z_e \odot h_{e-1} \qquad (9)$$

Equation 2: Time GRU

Dataset: Blackberry's QNX

- Data from QNX embedded in a vehicle
- Data logs consisted of over 1.000.000 events with 33 attributes each.
- Combined 'class' and 'event' attributes to redefine 93 different possible events.
- Previous attempts at predicting QNX traces have shown it is not a trivial task *(Lakhani et.al., 2019)*

time	class	event
24678463000	COMM	SND_PULSE_EXE
24678465041	KER_EXIT	TRACE_EVENT/01
24678479000	KER_CALL	MSG_SENDV/11
24678482208	COMM	SND_MESSAGE
24678492250	THREAD	THREPLY
24678494500	THREAD	THRUNNING
24678497708	COMM	REC_MESSAGE
24678499541	KER_EXIT	MSG_RECEIVEV/14
24678626833	KER_CALL	MSG_REPLYV/15
24678633166	COMM	REPLY_MESSAGE
24678635125	THREAD	THREADY
24678638083	KER_EXIT	MSG_REPLYV/15
24678656000	KER_CALL	MSG_RECEIVEV/14
24678661250	THREAD	THRECEIVE
24678664250	THREAD	THRUNNING
24678674458	COMM	REC_PULSE
24678676791	KER_EXIT	MSG_RECEIVEV/14
24678746166	INT_ENTR	0x00000029
24678747750	INT_HANDLER_ENTR	0x00000029
24678758583	COMM	SND_PULSE_EXE

Table 2: QNX event trace example

Workflow

- Model consisted of two RNN cells stacked together
- Events are fed as one-hot encoded vectors to the model
- Trace reconstruction ➔ Sequence prediction
- Given an input sequence of size N, the model predicts the upcoming sequence of events of size M

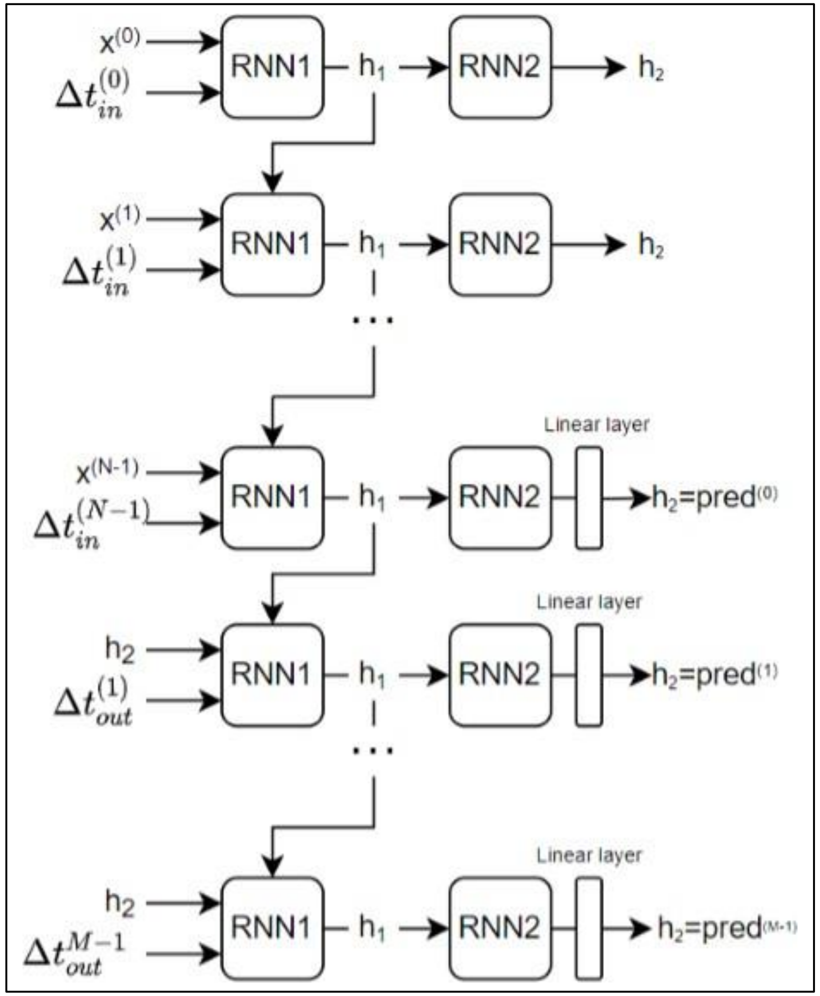

Figure 8: QNX Time-RNN modelling workflow

Experiments: Time GRU vs other RNNs

- Markov Chain as a baseline & 5 models: LSTM, GRU, Time LSTM 1, Time LSTM 2 and Time GRU
- Trained each model for 80 epochs on input sequences of 25 events.
- Report accuracy and Cohen's kappa coefficient.
- Time GRU is 25% faster than Time LSTM 1 and 43% faster than Time LSTM 2.

$$\kappa \equiv \frac{p_o - p_e}{1 - p_e} = 1 - \frac{1 - p_o}{1 - p_e}$$

Equation 3: Cohen's Kappa coefficient

Model	Output trace size			
	50 events		100 events	
	Acc.(%)	κ	Acc.(%)	κ
Markov	16.29	0.0522	14.32	0.0261
LSTM	50.38	0.4688	36.41	0.3087
GRU	47.19	0.4345	33.63	0.2765
Time LSTM 1	88.02	0.8727	87.13	0.8632
Time LSTM 2	88.10	0.8736	87.66	0.8688
Time GRU	86.58	0.8572	85.83	0.8493

Table 3: Time GRU vs other RNNs

Experiments: Discrete vs continuous time

- Instead of converting time intervals to vectors keeping them as scalars proved to improve modelling performance.
- Trained a Time GRU with continuous time representations for 50 epochs.

Accuracy: **93 %** *Cohen's Kappa coefficient:* **0.91**

Model	Output trace size			
	50 events		100 events	
	Acc.(%)	κ	Acc.(%)	κ
Markov	16.29	0.0522	14.32	0.0261
LSTM	50.38	0.4688	36.41	0.3087
GRU	47.19	0.4345	33.63	0.2765
Time LSTM 1	88.02	0.8727	87.13	0.8632
Time LSTM 2	88.10	0.8736	87.66	0.8688
Time GRU	86.58	0.8572	85.83	0.8493

Table 3: Time GRU vs other RNNs

Experiments: Time GRU generalization

- Trained Time GRU with continuous time on input and output sequences of 25 events.
- Evaluated its performance on different input/output size combinations
- Time GRU can even predict future sequences 20 times larger than the input sequences.

Results of trained model

Input size	Output size	Acc.	κ
25	25	92.54	0.9210

Validation of trained model

Input size	Output size	Acc.	κ
25	50	91.92	0.9144
25	100	91.18	0.9062
10	100	90.52	0.8993
10	200	89.58	0.8896
50	25	92.24	0.9178
100	25	92.13	0.9167

Table 4: Time GRU generalization

Conclusions

- Incorporating time into RNN models is crucial for real-life systems modelling
- Time GRUs can perform almost as good as Time LSTMs with a significantly higher speed.
- Time RNNs solve the problem of event sequence predictions of the QNX system.

References

S. Li, J. Bi, H. Yuan, M. Zhou, and J. Zhang, "Improved lstm-based prediction method for highly variable workload and resources in clouds," in2020 IEEE International Conference on Systems, Man,and Cybernetics (SMC). IEEE, 2020, pp. 1206–1211

Sucholutsky, A. Narayan, M. Schonlau, and S. Fischmeister, "Deep learning for system trace restoration," in2019 International Joint Conference on Neural Networks (IJCNN). IEEE, 2019, pp. 1–8

K. Mahato and A. Narayan, "Qmine: A framework for mining quantitative regular expressions from system traces," in2020 IEEE20th International Conference on Software Quality, Reliability and Security Companion (QRS-C). IEEE, 2020, pp. 370–377.

C. A. Bailer-Jones, D. J. MacKay, and P. J. Withers, "A recurrent neural network for modelling dynamical systems, "network: computation in neural systems, vol. 9, no. 4, pp. 531–547, 1998.

M. C. Mozer, D. Kazakov, and R. V. Lindsey, "Discrete event, continuous time rnns,"arXiv preprint arXiv:1710.04110, 2017.

Y. Zhu, H. Li, Y. Liao, B. Wang, Z. Guan, H. Liu, and D. Cai, "What to do next: Modeling user behaviors by time-lstm." in IJCAI, vol. 17,2017, pp. 3602–3608

K. Lakhani and A. Narayan, "A neural word embedding approach to system trace reconstruction," in2019 IEEE International Conference on Systems, Man and Cybernetics (SMC). IEEE, 2019, pp. 285–291.